A secure LoRaWAN sensor network architecture

Bogdan Oniga, Vasile Dadarlat
Department of Computer Sciences,
Technical University of Cluj-Napoca, Romania
Bogdan@Oniga.me, Vasile.Dadarlat@cs.utcluj.ro

Elie De Poorter
Gent University - imec
Gent, Belgium
Eli.DePoorter@UGent.be

Adrian Munteanu
Vrije Universiteit Brussel - imec
Brussels, Belgium
acmuntea@etrovub.be

***Abstract*—A novel secure architecture for sensor networks that make use of the LoRaWAN (Long-Range Wide Area Network) protocol specification is proposed in this paper. The paper analyses potential security threats and provides the protection mechanisms and security recommendations to enable protected data transmission and to prevent unauthorized access and data loss in LoRaWAN sensor networks.**

I. Introduction

LoRaWAN is a well-known low power wide area network technology that found a broad range of applications in IoT [1]. Providing secure data transmission and preventing security attacks in such applications is of critical importance. Security aspects in LoRaWAN have been addressed in the past in [2], which presents an analysis of the security of the Long Range (LoRa) solution and its LoRaWAN protocol. This work builds on the generic concepts of [2] and proposes a novel secure LoRaWAN sensor architecture. The paper analyses the potential security threats and implements different security controls in order to protect data transmitted over the network and to establish a strong line of defense in such networks.

II. Security mechanisms in LoRaWAN

LoRaWAN is a protocol specification built on top of the LoRa technology to enable low power, secure, bi-directional, wide-area communication between remote sensors and gateways connected to the network [1].

LoRaWAN provides encryption and signing of packets sent over the network. For encryption, symmetric keys known by the End-nodes, Network and Application Servers are used, these being distributed in two different ways, depending on the employed activation method, namely Over-The-Air Activation (OTAA) and Activation By Personalization (ABP) [1].

Both activation methods provide a strong level of security using symmetric encryption in message exchanges between End-nodes and servers. The provided mechanisms eliminate any possibility of an attacker to inject malicious End-nodes or to take advantage of the network in the activation process without knowing the End-node's keys.

Despite of these built-in security mechanisms, a LoRaWAN network should also be a *secure network architecture*, which applies security controls and monitoring of the network to provide confidentiality, data integrity and availability. In this context, a novel secure LoRaWAN sensor network architecture is proposed, as detailed next.

III. Proposed Secure LoRaWAN architecture

The proposed architecture, depicted in Fig. 1, implements a Public Key Infrastructure (PKI) model using the open-source solution provided by Cloudflare [3]. In this model, each component holds a certificate signed by its own Certificate Authority (CA) implemented at network level. All entities use these certificates to communicate with relevant entities in the network. This approach provides encryption, authentication and integrity in message exchanges.

A. Network traffic control

For improved security, each entity implements Iptables rules which allow only network traffic necessary for a functional LoRaWAN application and for system maintenance, any other unnecessary connections being denied.

B. Gateway-network communication

As shown in Fig. 1, the Gateway cannot be considered a trustworthy network entity, as it lies outside of the controlled network area. To provide secure data transmission and alleviate potential exposure to security attacks originating from the Gateway, the proposed LoRaWAN network implements a Remote Access Virtual Private Network (VPN) server. In order to access the internal network, each Gateway holds a certificate generated by the VPN server, providing Gateway authentication and preventing any Gateway impersonation attempts. Also, the VPN server maintains a white list of Gateway IPs, avoiding in this manner the access of potentially malicious Gateways or Gateway replication. The software utilized for implementing the VPN server is OpenVPN [4].

Once connected to the internal network, the Gateways should exchange data only with Brokers in a secure way, over Datagram Transport Layer Security (DTLS) - see Fig. 1; this provides security for User Datagram Protocol (UDP) communications and prevents attacks such as eavesdropping, tampering, or message forgery.

C. Broker level

The Broker's role is to transport End-node messages received from Gateways to upper network levels using MQTT messaging protocol. To provide a secure connection between the MQTT server, implemented at Broker level, and the Network Server, the message exchanges should be made over Transport Layer Security (TLS), a protocol that is capable of securing the transport over TCP.

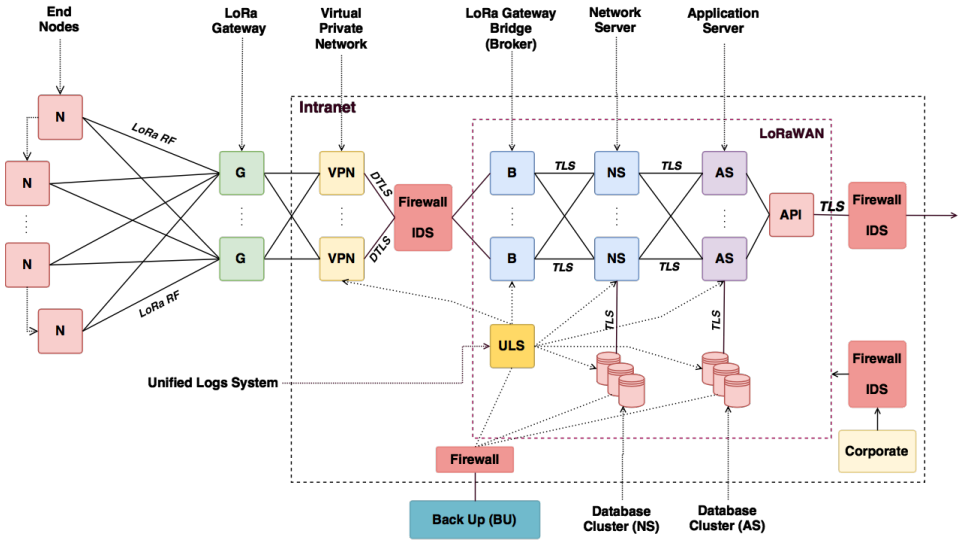

Fig. 1. Proposed LoRaWAN secure network architecture.

D. Network Server level

The Network Server interacts with multiple entities such as the Broker, Application Server and the database (see Fig. 1). The Network Server is responsible for network control and MAC commands, main actions being to manage End-node sessions, schedule acknowledgements, manage receiving windows of End-nodes, eliminate duplicate packets and adapt data rates. Moreover, the Network Server uses the network keys (NwkSKey) to sign and verify messages sent between End-nodes and the server in order to prevent data processing of altered messages. Therefore, the key management mechanism should be carefully implemented as the Network Server represents a single point of failure in the LoRaWAN network.

E. Application Server level

The Application Server is also a sensitive component as it generates and stores the keys used by the End-nodes in the activation process and message encryption. The keys must be stored in a secure way and accessible only by the Application Server, the exception being the network keys (NwkSKey), which have to be transmitted to the Network Server.

The Application Server employs also an Application Programming Interface (API), which is used for application and End-node management and to expose End-node data. This interface is an important component as it provides access to sensitive data outside of the controlled network area to applications making use of these data. Usually, the API exposes data through a web server. Hence, the web server must implement a secure connection by using Hyper Text Transfer Protocol Secure (HTTPS), based on a TLS certificate signed by a global Certificate Authority (e.g. Comodo or Let's Encrypt).

The payloads received from End-nodes must be validated and sanitized by the Application Server before performing any other operations. The Application Server must implement functions that check if the payload meets a set of criteria (e.g. testing for length, format, range, and allowable characters), accepting only expected payload formats. These techniques are used to provide an in-depth defense at application level.

F. Access management

Access management is an important aspect in a secure infrastructure being responsible to granting access and privileges to authorized users. Each entity, part of the proposed infrastructure, should allow a Secure Shell (SSH) connection, which is necessary for system updating and maintenance processes. The access should be made only from the internal network for authorized persons based on user and IP whitelisting.

G. Availability

Availability aims at maintaining a correct functioning of the system in order to keep data and resources available for authorized use. To enable availability, the following mechanisms were proposed and practically implemented.

Load Balancing, which is used to distribute the network traffic to multiple services. Load balancers increase the capacity of a system to support concurrent connections and keep data transmission continuity between components even if one or more services are facing functional failures. In the proposed secure architecture, load balancing can be used for each entity, including the VPN, the Broker, the Network and the Application Servers, and the API.

Database Clustering. Database clustering is a solution used to increase database availability. In the proposed LoRaWAN network architecture, two database clusters are present, serving the Network and Application Servers respectively. It is mandatory to use database clustering and to develop fail-over strategies for each database in the network in order to provide continuity and availability of data transmission.

Data Backups. In the proposed architecture, data backups are required for the databases used by the Network and Application Servers. Data backups are also required for the

TABLE I
TESTING SCENARIOS, RESULTS AND RECOMMENDATIONS.

Entity	Scenario	Result	Recommendation
End-node	Sniffing LoRa Traffic	LoRaWAN messages are encrypted using AES128 offering a strong protection against intercepting data in transit.	Keys should be securely stored and access should be allowed only to authorized entities.
End-node	End-node impersonation	LoRaWAN offers strong protection against impersonation based on the sessions handled by the server. Impersonation attempts generate error messages at Network Server.	Implement a monitoring mechanism that alerts suspicious activities based on messages received at Network Server level.
End-node	Replay Messages	Replay attacks are feasible in ABP, only OTAA configuration offers protection against such attacks.	Implement OTAA configuration for End-nodes.
Gateway	Manipulate communication's parameters	Manipulating the communication parameters may lead to high power consumption on the End-nodes, battery depletion, sensor breakdown and higher costs.	Prevent unauthorized access to the Gateway.
Gateway	Man-in-the-middle attack performed between Gateway and Broker	In this case, an attacker is able to alter the communication between the Gateway and the internal network components, to intercept all messages or even inject new ones.	Implement a VPN server which provides encryption and authentication for Gateways. Firewall rules and IDS are necessary to enhance security.
IDS	Rules violation	Test the logging and alerting mechanisms by thoroughly testing the implemented IDS rules	Employ security and event-management tools to centralize and prioritize the alerts.
Network Server	Denial of Service: database Out-of-Memory	The Network Server creates new sessions and keeps the old ones for each End-node. An attack generating a large number of sessions may lead to database out-of-memory.	Remove any old End-node session when a new session is generated. Also, each session should have assigned an expiration time.
Application Server	Sending malicious payloads using the End-nodes	No vulnerabilities found, but may differ from one implementation to another.	The Application Server should only accept expected payload formats from the End-nodes.

Unified logging system (see Fig. 1), which preserves the log's of each network entity. The data backups should be preserved in a secure environment, outside of the internal network.

H. Monitoring and Intrusion detection system (IDS)

In a secure architecture model, the presence of an intrusion detection system is required to monitor network traffic for malicious activity or policy violations. In the proposed architecture, the IDS implementation is based on Snort [5], which is an open source intrusion detection system for real-time traffic analysis and packet logging on IP networks.

In a LoRaWAN network, the most sensible points are the entry points of the internal network; at these points, one has to implement firewall rules to restrict connections other than authorized connections required by the application's functionalities.

In a LoRaWAN network, the traffic to and from the Gateways can be predicted by calculating the traffic generated by the End-nodes. A good practice is to implement an anomaly-based intrusion detection technique which monitors the traffic and compares it against established baselines. As baselines one identifies the usual bandwidth (with some margins), the employed protocols, usually connected ports and devices; such a system generates alerts when the detected traffic is significantly different than the baseline.

Another sensitive points of the network are the API and the access from corporate to the internal network. Network traffic control is provided by the implemented Firewall rules. These connection should be monitored by implementing a signature-based intrusion detection technique which compares network packets against a database of signatures or attributes from known malicious threats. If malicious threats or unauthorized access attempts to the internal network's entities are detected, the system generates alerts and notifies the network administrators. An anomaly-based detection technique can also be used to complement the traffic monitoring process.

IV. SECURITY RECOMMENDATIONS

During security assessment of the proposed LoRaWAN network architecture, multiple testing scenarios were performed. Table I presents testing results and recommendations concluded from each scenario. Details about the security aspects of LoRa as well as more testing and implementation details of this architecture are given in our recent work in [6].

V. CONCLUSIONS

The standard security features provided by LoRaWAN are not sufficient for secure end-to-end sensor networks and applications. This paper addresses the security concerns of data protection and data privacy in applications based on LoRaWAN. The paper proposes and implements a secure LoRaWAN network architecture and presents a set of best practices that have to be followed in order to build end-to-end secure LoRaWAN sensor networks.

ACKNOWLEDGMENTS

The research work was funded by FWO-Flanders, projects G084117 and G025615.

REFERENCES

[1] LoRa Alliance (2015) LoRa specification. [Online] http://bit.ly/LoRaWAN-specification
[2] R. Miller. (2016, Mar.) LoRa Security Building a Secure LoRa Solution. [Online] https://labs.mwrinfosecurity.com/assets/BlogFiles/mwri-LoRa-security-guide-1.2-2016-03-22.pdf
[3] Cloudflare SSL software repository. [Online] https://github.com/cloudflare/cfssl
[4] OpenVPN website. [Online] https://openvpn.net/
[5] Snort website. [Online] https://snort.org/
[6] B. Oniga, V. Dadarlat, E. De Poorter, A. Munteanu, "Analysis, design and implementation of secure LoRaWAN sensor networks," IEEE Int. Conf. Intel. Computer Commun. and Proc., ICCP 2017, Romania, Sept. 2017.